

Documento di Specifica dei Requisiti Software

Documento contenente le specifiche dei requisiti funzionali e non funzionali per il progetto di Gestione di una Rubrica Telefonica.

0612707416 - Gioia Iannuzzi
0612708273 - Luigi Montonetti
0612707862 - Arianna Paletta
0612708294 - Debora Villano

Indice

1. Introduzione
 - 1.1. Propositi
 - 1.2. Scopo
 - 1.3. Pubblico Destinatario
 - 1.4. Convenzioni del Documento
2. Descrizione Generale
 - 2.1. Prospettiva del Prodotto
 - 2.2. Funzionalità del Prodotto
 - 2.3. Utenti e Caratteristiche
 - 2.4. Vincoli di Design e Implementazione
 - 2.5. Assunzioni e Dipendenze
3. Requisiti Specifici
 - 3.1. Requisiti di Interfaccia Esterna
 - 3.2. Requisiti Funzionali
 - 3.3. Descrizione dei Casi d'Uso
 - 3.3.1. Diagramma UML dei Casi d'Uso
4. Requisiti Non Funzionali e Attributi di Qualità
 - 4.1. Requisiti di Prestazione
 - 4.2. Requisiti di Sicurezza e Protezione
 - 4.3. Attributi di Qualità

1 Introduzione

1.1 Propositi

Il proposito di questo documento è specificare i requisiti del sistema software *Rubrica Telefonica*, con l'obiettivo di guidarne la progettazione, realizzazione e validazione. Il documento è redatto seguendo le linee guida stabilite nello standard *IEEE Std 830-1993* (IEEE Recommended Practice for Software Requirements Specifications), per garantire che i requisiti siano chiari, completi e facilmente comprensibili da tutti gli stakeholder coinvolti nel progetto.

1.2 Scopo

Il prodotto software descritto in questo documento è una **rubrica telefonica**, progettata per la gestione ed organizzazione di numeri telefonici. L'applicazione sarà progettata per funzionare su sistemi operativi quali: Windows, macOS e Linux, con funzionalità di ricerca, modifica e organizzazione.

1.3 Pubblico Destinatario

Il pubblico destinatario di questo documento include le persone coinvolte nella progettazione, sviluppo, utilizzo e manutenzione del sistema:

- **Sviluppatori:** Responsabili della progettazione, implementazione e manutenzione del software, che utilizzeranno il documento per comprendere i requisiti, le specifiche tecniche del sistema e il flusso di progettazione corretto da seguire.
- **Tester:** I professionisti che verificheranno il corretto funzionamento del sistema, utilizzando questo documento come riferimento.
- **Utenti Finali:** Le persone che utilizzeranno l'applicazione per gestire i propri contatti. Coloro che potrebbero essere coinvolti nel processo di feedback durante lo sviluppo.
- **Amministratori di Sistema:** Responsabili della gestione dell'infrastruttura, della sicurezza e della manutenzione del sistema, nonché del supporto agli utenti.

1.4 Convenzioni del Documento

Simboli e Terminologia

- SRS: sta per Software Requirements Specifications.
- Acronimi utilizzati per la categorizzazione dei requisiti:
 - IF: Funzionalità Individuale
 - BF: Business Flow
 - DF: Dato e Formato dei Dati
 - UI: Interfaccia Utente
 - IS: Interfaccia con altri Sistemi
 - FC: Ulteriori Vincoli
 - NF: Non Funzionale

Requisiti e Scenari Gli scenari (IF, BF, DF, UI) fungono da indice e permettono di collegare ciascun requisito al relativo flusso o funzionalità. Se un requisito non è collegato ad alcun *business flow*, sarà identificato da un solo pedice (ad esempio, **IS-1** o **FC-1**).

Classificazione delle Priorità La priorità dei requisiti è suddivisa in:

- **Alta:** Requisiti critici per il funzionamento dell'applicazione.
- **Media:** Requisiti utili ma non essenziali, implementabili in una fase successiva.
- **Bassa:** Requisiti opzionali o di minor impatto.

Valutazione del Rischio Tecnico Il rischio tecnico dei requisiti viene classificato come:

- **Basso:** Implementazione semplice e priva di complessità.
- **Medio:** Richiede un moderato sforzo tecnico o risorse specifiche.
- **Alto:** Potrebbe causare problemi significativi o richiedere tecnologie avanzate.

Acronimi Utilizzati

- **GUI** (*Graphical User Interface*): Interfaccia Grafica Utente.
- **CSV** (*Comma-Separated Values*): Formato di file utilizzato per memorizzare dati tabellari in formato testuale, separati da virgole.

Casi d'Uso La spiegazione dei casi d'uso segue il modello strutturato riportato di seguito:

- **Nome del Requisito:** Titolo identificativo.
- **Obiettivo:** Scopo del caso d'uso.
- **Attori:** Persone o sistemi coinvolti.
- **Pre-condizioni:** Requisiti da soddisfare prima di eseguire il caso d'uso.
- **Post-condizioni:** Risultato atteso dopo l'esecuzione del caso d'uso.
- **Input:** Dati o informazioni necessarie per l'esecuzione.
- **Flusso Normale:** Sequenza di eventi standard.
- **Flusso Alternativo:** Varianti o eccezioni al flusso normale.

2 Descrizione Generale

2.1 Prospettiva del Prodotto

L'applicazione che si vuole realizzare è un sistema software autonomo, indipendente da prodotti esterni. Tuttavia, è in grado di integrarsi con questi, garantendo una gestione sicura e accessibile dei dati degli utenti.

2.2 Funzionalità del Prodotto

Le principali funzionalità del prodotto includono:

- **Creazione, Modifica ed Eliminazione dei Contatti:** Consente agli utenti di mantenere una lista aggiornata dei propri contatti.
Dando la possibilità di inserire nuovi contatti nella propria rubrica telefonica, di modificarne o eliminarne alcuni tra quelli già presenti. Sono liberi di aggiungere dettagli come il nome completo del contatto, il numero di telefono e l'indirizzo email, oltre a eventuali note personalizzate.
- **Visualizzazione Ordinata:** I contatti sono elencati in ordine alfabetico per rendere più facile la consultazione.
- **Funzionalità di Ricerca Avanzata:** Permette di trovare i contatti utilizzando il nome completo o parziale, il numero di telefono oppure l'indirizzo email, tramite una ricerca che non fa distinzione tra maiuscole e minuscole.
- **Gestione dei Duplicati:** Il sistema rileva e gestisce i duplicati di contatti, chiedendo conferma per evitare conflitti.
- **Backup e Ripristino:** Un'opzione per salvare i dati dell'utente e ripristinarli se necessario, al fine di mantenere al sicuro le informazioni sensibili.
- **Personalizzazione della Rubrica:** Gli utenti hanno la possibilità di annotare ulteriori dati sui contatti, come date di compleanno, relazioni tra i contatti e altre informazioni.

2.3 Utenti e Caratteristiche

Gli utenti *della Rubrica Telefonica* sono divisi in due categorie:

- **Utenti Finali:** Persone che utilizzano l'applicazione per gestire i propri contatti. Gli utenti finali hanno bisogno di un'interfaccia semplice e intuitiva, che consenta di aggiungere, modificare e consultare rapidamente i contatti.
- **Amministratori:** Gli amministratori sono incaricati della configurazione, gestione e della manutenzione generale dell'applicazione.

Il punto di forza del software è la sua facilità d'uso, che lo rende accessibile a utenti con diverse competenze tecniche, inclusi coloro che non hanno esperienza nell'uso di applicazioni simili.

2.4 Vincoli di Design e Implementazione

Standard Tecnici

Il linguaggio di programmazione utilizzato per il software della rubrica telefonica è **Java 8** con l'ausilio di **JavaFX** per la realizzazione dell'interfaccia utente.

Vincoli

Il sistema deve essere in grado di rispondere entro un secondo alla maggior parte delle richieste effettuate dall'utente, come l'aggiunta, la modifica o la ricerca di un contatto. I file di salvataggio devono essere generati e letti in formati standard come **CSV** o **JSON** per garantire l'interoperabilità con altri software.

Limitazioni sui Dati

Il sistema non permetterà di aggiungere più di **3 numeri di telefono** e più di **3 indirizzi e-mail** per ciascun contatto.

Interfaccia

L'interfaccia utente deve essere pratica e facile da utilizzare anche per utenti non esperti. Comprenderà un design minimalista con pochi pulsanti.

Vincoli per il Rilascio

Il sistema deve essere pronto per il rilascio entro il **15 Dicembre**.

2.5 Assunzioni e Dipendenze

Ambiente di Utilizzo

La rubrica telefonica sarà utilizzabile esclusivamente su **computer** (desktop o laptop) e non è prevista una versione per dispositivi mobili o web.

Competenze Base

Si assume che gli utenti abbiano una conoscenza di base sull'utilizzo di un computer, inclusa la capacità di interagire con un'interfaccia grafica e utilizzare applicazioni di uso comune.

Dati

Si assume che gli utenti inseriscano dati corretti per ogni contatto. Sarà implementata una funzionalità di validazione per i numeri di telefono e gli indirizzi e-mail, al fine di garantire il corretto formato dei dati inseriti.

Database

Il sistema software dipenderà dall'integrità dei dati che verranno memorizzati all'interno di un **database**, il quale deve essere sicuro e protetto da possibili errori di corruzione dei dati.

3 Requisiti Specifici

3.1 Requisiti di Interfaccia Esterna

Disponibilità di un'interfaccia grafica (GUI) Il sistema deve offrire un'interfaccia grafica intuitiva. Questo fa sì che gli utenti possano navigare senza una formazione avanzata.

Accessibilità multi-dispositivo Il sistema deve essere accessibile su vari Sistemi Operativi. Questo garantisce un'esperienza di navigazione a più utenti.

3.2 Requisiti Funzionali

I requisiti funzionali descrivono le operazioni principali che il sistema deve essere in grado di offrire per soddisfare le esigenze degli utenti e raggiungere gli obiettivi. Ogni requisito è identificato da un ID (vedi paragrafo 1.4), e classificato per priorità, descrizione e rischio tecnico. Questo per una facile comprensione da parte del cliente.

ID	Requisito	Priorità	Breve descrizione	Rischio tecnico
IF-1.1	Creare Contatti	Alta	Deve essere possibile aggiungere nuovi contatti alla rubrica.	Basso
IF-1.2	Modificare Contatti	Alta	È possibile modificare i dettagli di un contatto esistente.	Basso
IF-1.3	Cancellare Contatti	Alta	È possibile rimuovere contatti dalla rubrica chiedendo una conferma.	Basso
IS-1	Salvare e Caricare Contatti	Alta	Le informazioni della rubrica devono essere salvate su file e caricate da file, mantenendo l'intera rubrica.	Medio
BF-1	Validazione dei campi obbligatori	Alta	È obbligatorio inserire un nome e/o un cognome.	Medio
DF-1	Nome e Cognome	Alta	Ogni contatto deve avere un nome e un cognome, ma almeno uno dei due deve essere presente.	Basso
DF-2	Numeri di Telefono	Alta	Ogni contatto può avere da 0 a 3 numeri di telefono.	Basso
IF-2.1	Ricerca di Contatto	Media	È possibile cercare un contatto inserendo la sottostringa iniziale di nome, cognome, numero o mail.	Medio
UI-1	Aggiunta Note di un Contatto	Bassa	È possibile aggiungere delle note ad un contatto.	Medio
IF-2.2	Visualizzazione Contatti	Alta	È possibile visualizzare una lista dei contatti.	Medio

BF-2	Gestione Duplicati	Media	Il sistema richiede conferma per contatti con lo stesso nome, permettendo di sovrascrivere o creare un nuovo contatto con dettagli aggiuntivi.	Medio
UI-2	Aggiunta Foto Contatto	Bassa	È possibile aggiungere una foto ad un contatto.	Alto
UI-3	Interfaccia Grafica	Alta	Il sistema deve disporre di un'interfaccia grafica.	Basso
UI-4	Tastierino di Aggiunta Contatto	Bassa	È possibile salvare un contatto digitando il numero di telefono da tastierino.	Alto
IF-2.3	Menu Preferiti	Media	L'utente può aggiungere contatti alla lista dei preferiti e accedervi rapidamente.	Medio

Tabella 1: Tabella di Categorizzazione dei Requisiti Funzionali

3.3 Descrizione dei Casi d'Uso

Creazione Contatto

Campo	Descrizione
Obiettivo	L'utente può aggiungere contatti alla rubrica.
Attori partecipanti	Utente
Pre-condizioni	Deve essere accessibile l'interfaccia per la creazione del contatto.
Post-condizioni	Il contatto è salvato nel sistema ed è visibile nella rubrica tramite operazioni di ricerca o visualizzazione.
Input	<ul style="list-style-type: none">• Nome• Cognome• Numeri di telefono (massimo 3)• E-mail (massimo 3)• Note (opzionale)• Foto (opzionale)
Flusso normale	<ol style="list-style-type: none">1. L'utente accede alla schermata di creazione di un nuovo contatto.2. L'utente inserisce i dati richiesti (nome, cognome, numero di telefono e/o email).3. L'utente clicca su <i>Salva</i>.4. Il sistema convalida i dati.5. Il contatto viene salvato nella memoria.6. Il sistema conferma che l'operazione è avvenuta con successo.

Flusso alternativo 1	<p>2a. L'utente non compila tutti i campi richiesti</p> <p>2a.1. L'utente prova a salvare il contatto senza compilare i campi richiesti.</p> <p>2a.2. Il sistema mostra un messaggio di errore indicando che deve essere inserito almeno uno tra <i>nome</i> o <i>cognome</i>.</p> <p>2a.3. L'utente può correggere l'errore.</p>
Flusso alternativo 2	<p>2b. Inserimento di formati non validi nel campo di ricerca</p> <p>2b.1. L'utente inserisce dati con formati non validi (ad esempio una stringa di caratteri nel campo <i>numero di telefono</i>).</p> <p>2b.2. Il sistema mostra un messaggio di errore indicando il formato corretto dei campi.</p> <p>2b.3. L'utente può correggere l'errore.</p>
Flusso alternativo 3	<p>3a. L'utente prova a salvare un contatto già esistente</p> <p>3a.1. Il sistema mostra un messaggio di errore indicando l'eventuale duplicazione.</p> <p>3a.2. L'utente può aggiornare il contatto già esistente o annullare l'operazione.</p>

Modifica Contatto

Campo	Descrizione
Obiettivo	L'utente può modificare i dettagli di un contatto esistente nella rubrica.
Attori partecipanti	Utente
Pre-condizioni	<ul style="list-style-type: none">• La rubrica deve contenere almeno un contatto.• Il contatto da modificare deve essere individuabile tramite ricerca.
Post-condizioni	<ul style="list-style-type: none">• I dati di un contatto vengono aggiornati con nuovi dati forniti dall'utente.• Il contatto aggiornato è visibile nella rubrica.
Input	<ul style="list-style-type: none">• Nome• Cognome• Sotto-stringa del nome• Sotto-stringa del cognome• Numero di telefono• E-mail• Dati aggiornati che l'utente vuole modificare

Flusso normale	<ol style="list-style-type: none"> 1. L'utente accede alla rubrica e cerca il contatto da modificare. 2. L'utente seleziona l'opzione modifica. 3. L'utente modifica uno o più campi. 4. L'utente clicca su "Salva". 5. Il sistema convalida i dati. 6. Il contatto viene salvato nella memoria. 7. Il sistema conferma che l'operazione è avvenuta con successo.
Flusso alternativo 1	<p>1a. Contatto non presente in rubrica:</p> <p>1a.1. Il sistema mostra un messaggio di errore indicando che il contatto cercato non è presente.</p>
Flusso alternativo 2	<p>3a. Modifiche con formati non validi</p> <p>3a.1. Il sistema mostra un messaggio di errore indicando il formato corretto dei campi.</p> <p>3a.2. L'utente può correggere l'errore.</p>
Flusso alternativo 3	<p>4a. Nessuna modifica effettuata:</p> <p>4a.1. Il sistema mostra un messaggio di errore indicando che non è stata effettuata alcuna modifica.</p> <p>4a.2. L'utente può scegliere se modificare o uscire dall'opzione di modifica.</p>
Flusso alternativo 4	<p>4b. Contatto duplicato dopo la modifica:</p> <p>4b.1. Il sistema mostra un messaggio di errore indicando la duplicazione.</p> <p>4b.2. L'utente può unire i contatti duplicati o annullare l'operazione.</p>

Eliminazione contatto

Campo	Descrizione
Obiettivo	L'utente può eliminare un contatto esistente dalla rubrica.
Attori partecipanti	Utente
Pre-condizioni	<ul style="list-style-type: none">• La rubrica deve contenere almeno un contatto.• Il contatto da eliminare deve essere individuabile tramite ricerca.
Post-condizioni	<ul style="list-style-type: none">• Il contatto selezionato dall'utente viene rimosso dalla rubrica.• Il contatto rimosso non è più visibile.
Input	<ul style="list-style-type: none">• Nome• Cognome• Sotto-stringa del nome• Sotto-stringa del cognome• Numero di telefono• E-mail
Flusso normale	<ol style="list-style-type: none">1. L'utente accede alla rubrica e cerca il contatto da eliminare.2. L'utente seleziona l'opzione "Elimina".3. Il sistema chiede una conferma per l'eliminazione.4. L'utente conferma l'operazione.5. Il sistema rimuove il contatto dalla rubrica e dalla memoria.6. Il sistema conferma che l'operazione è avvenuta con successo.

Flusso alternativo 1	1a. Contatto non presente in rubrica: 1a.1. L'utente cerca di eliminare un contatto non presente. 1a.2. Il sistema mostra un messaggio di errore indicando che il contatto non è presente nella rubrica.
Flusso alternativo 2	1b. Inserimento di formati non validi nel campo ricerca 1b.1. Il sistema mostra un messaggio di errore e richiede l'inserimento di criteri validi.. 1b.2. L'utente corregge i criteri e riprende il flusso normale.
Flusso alternativo 3	3a. Annullamento dell'operazione: 3a.1. L'utente seleziona "Elimina", ma decide di annullare l'operazione. 3a.2. Il sistema non rimuove il contatto.

Salvataggio e Caricamento Contatti

Campo	Descrizione
Obiettivo	L'utente può salvare i contatti della rubrica in un file o database per conservarli in modo sicuro.
Attori partecipanti	Utente
Pre-condizioni	<ul style="list-style-type: none">• La rubrica deve contenere almeno un contatto.• Il percorso di salvataggio deve essere accessibile.
Post-condizioni	<ul style="list-style-type: none">• I contatti vengono salvati nel formato scelto (es. CSV, JSON, ecc.).
Input	<ul style="list-style-type: none">• Rubrica da salvare• Formato in cui salvare la rubrica
Flusso normale	<ol style="list-style-type: none">1. L'utente accede alla rubrica.2. L'utente seleziona l'opzione "Salva contatti".3. Il sistema chiede il percorso su cui salvare i contatti.4. L'utente clicca su "Salva".5. Il sistema esegue il salvataggio.6. Il sistema conferma che l'operazione è avvenuta con successo.
Flussi alternativo 1	<p>3a. Il percorso selezionato non è valido.</p> <p>3a.1. Il sistema non riesce ad accedere al percorso specificato e mostra un messaggio di errore.</p> <p>3a.2. L'utente seleziona un nuovo percorso valido.</p> <p>3a.3. Il sistema riprende il flusso normale e completa il salvataggio.</p>

<p>Flusso alternativo 2</p>	<p>5a. Si verifica un errore durante il salvataggio (es. mancanza di spazio o inserimento di dati con formati non validi).</p> <p>5a1. Il sistema mostra un messaggio di errore indicando la causa dell'impossibilità di salvataggio.</p> <p>5a2. L'utente può riprovare il salvataggio, scegliendo un percorso alternativo o liberando spazio.</p> <p>5a3. L'utente può anche decidere di annullare l'operazione.</p>
------------------------------------	---

Ricerca Contatto

Campo	Descrizione
Obiettivo	L'utente può cercare un contatto specifico nella rubrica, basandosi su informazioni come nome, cognome, numero di telefono o indirizzo e-mail.
Attori partecipanti	Utente
Pre-condizioni	Deve essere accessibile l'interfaccia per la ricerca dei contatti.
Post-condizioni	<ul style="list-style-type: none"> • Vengono visualizzati i contatti corrispondenti ai criteri di ricerca.
Input	<ul style="list-style-type: none"> • Nome • Cognome • Numero di telefono • E-mail
Flusso normale	<ol style="list-style-type: none"> 1. L'utente accede alla schermata di ricerca dei contatti. 2. L'utente inserisce le informazioni. 3. L'utente clicca su "Cerca". 4. Il sistema restituisce i risultati della ricerca. 5. L'utente può visualizzare e selezionare uno dei risultati.
Flusso alternativo 1	<p>1b. L'utente inserisce formati non validi nel campo ricerca</p> <p>1b.1. Il sistema mostra un messaggio di errore e richiede l'inserimento di criteri validi..</p> <p>1b.2. L'utente corregge i criteri e riprende il flusso normale.</p>
Flusso alternativo 2	<p>3b. L'utente inserisce informazioni non presenti nella rubrica.</p> <p>3b.1. Il sistema mostra un messaggio di errore indicando che il contatto non è presente in rubrica.</p>

Gestione del *Menù Preferiti*

Campo	Descrizione
Obiettivo	L'utente può aggiungere, visualizzare e gestire i contatti in un menu per un accesso rapido.
Attori partecipanti	Utente
Pre-condizioni	Deve essere accessibile l'interfaccia per la gestione dei preferiti.
Post-condizioni	<ul style="list-style-type: none"> • I contatti selezionati sono visibili nel menu preferiti. • L'utente può accedere rapidamente ai contatti preferiti.
Input	<ul style="list-style-type: none"> • Contatti da aggiungere ai preferiti
Flusso normale	<ol style="list-style-type: none"> 1. L'utente seleziona o cerca un contatto da aggiungere nel menu preferiti. 2. L'utente clicca su Aggiungi ai preferiti. 3. Il sistema convalida i dati. 4. Il contatto viene salvato nel menu preferiti. 5. L'utente può visualizzare l'elemento nel menu preferiti.
Flusso alternativo 1	<p>2a. L'utente cerca di aggiungere un contatto già presente nei preferiti.</p> <p>2a.1. Il sistema mostra un messaggio di errore indicando che l'elemento è già presente.</p>
Flusso alternativo 2	<p>2b. L'utente rimuove un contatto dai preferiti.</p> <p>2b.1. L'utente seleziona un contatto e clicca su <i>Rimuovi dai preferiti</i>.</p> <p>2b.2. Il sistema rimuove l'elemento dal menu preferiti.</p>

<p>Flusso alternativo 3</p>	<p>3b. Inserimento di formati non validi nel campo ricerca</p> <p>3b.1. Il sistema mostra un messaggio di errore e richiede l'inserimento di criteri validi..</p> <p>3b.2. L'utente corregge i criteri e riprende il flusso normale.</p>
------------------------------------	---

Visualizzazione Contatto

Campo	Descrizione
Obiettivo	L'utente può visualizzare i dettagli di un contatto presente nella rubrica.
Attori partecipanti	Utente
Pre-condizioni	<ul style="list-style-type: none">• La rubrica deve contenere almeno un contatto.• Il contatto da visualizzare deve essere individuabile tramite ricerca o selezione.
Post-condizioni	<ul style="list-style-type: none">• L'utente può consultare tutte le informazioni associate al contatto selezionato.
Input	<ul style="list-style-type: none">• Nome• Sottostringa del nome• Cognome• Sottostringa del cognome• Numero di telefono• E-mail

Flusso normale	<ol style="list-style-type: none"> 1. L'utente inserisce i criteri di ricerca (es. nome, cognome o altro identificatore). 2. Il sistema ricerca e presenta l'elenco dei contatti corrispondenti ai criteri inseriti. 3. L'utente seleziona il contatto desiderato dall'elenco. 4. Il sistema mostra i dettagli completi del contatto, inclusi: <ul style="list-style-type: none"> • Nome • Sottostringa del nome • Cognome • Sottostringa del cognome • Numeri di telefono • Indirizzi E-mail • Note (se presenti) • Foto (se presente)
Flussi alternativi	<p>1a. L'utente cerca un contatto non presente.</p> <p>1a.1. Il sistema mostra un messaggio di errore indicando che il contatto non è presente nella rubrica.</p>
Flusso alternativo 2	<p>1b. Inserimento di criteri di ricerca non validi</p> <p>1b.1. Il sistema mostra un messaggio di errore e richiede l'inserimento di criteri validi..</p> <p>1b.2. L'utente corregge i criteri e riprende il flusso normale.</p>

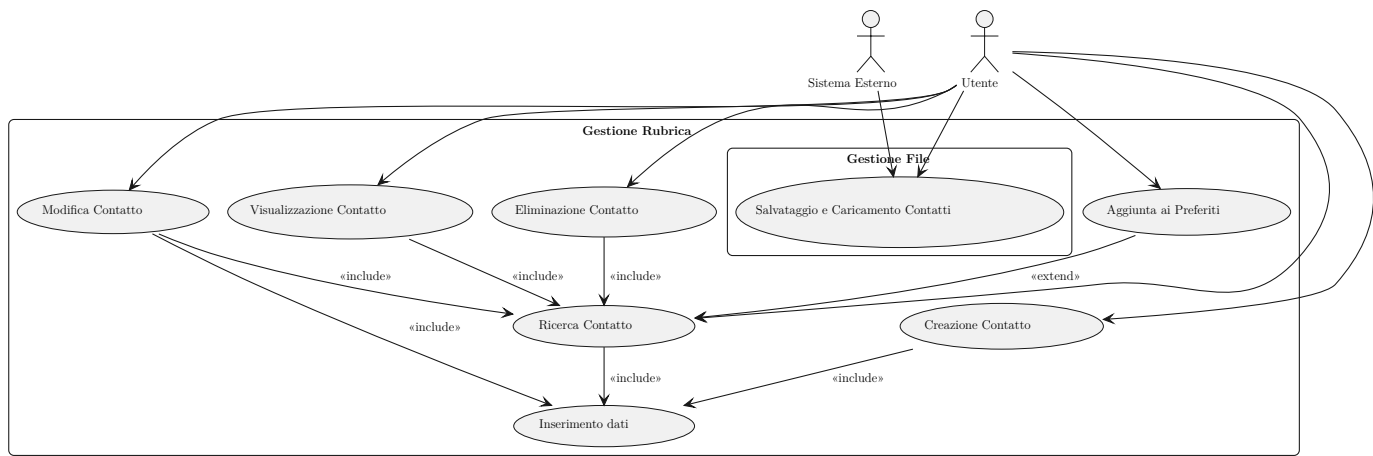


Figura 1: Diagramma UML dei Casi d'Uso
Realizzato con PlantUML

4 Requisiti Non Funzionali e Attributi di Qualità

In questa sezione vengono descritti i requisiti non funzionali e gli attributi di qualità che il sistema deve soddisfare per garantire efficienza, sicurezza e un'esperienza utente ottimale. Questi requisiti definiscono le prestazioni attese, le misure di sicurezza e gli attributi di qualità che influiscono sulla manutenibilità, modularità ed efficacia complessiva del sistema. Tali aspetti sono fondamentali per assicurare non solo il corretto funzionamento del software, ma anche la sua capacità di adattarsi e rispondere alle esigenze degli utenti e degli sviluppatori nel tempo.

4.1 Requisiti di Prestazione

- **Ricerca di un contatto:** La ricerca di un contatto deve avvenire in meno di un secondo.
- **Modifica di un contatto:** La modifica di un contatto deve avvenire in maniera rapida, in meno di un secondo.
- **Creazione di un contatto:** La creazione di un contatto deve avvenire in meno di un secondo.
- **Velocità caricamento:** L'interfaccia grafica dell'utente deve aprirsi in maniera rapida senza scatti.
- **Interazione:** L'interazione con le funzionalità dell'interfaccia grafica deve essere fluida e non causare ritardi.
- **Gestione della memoria:** I dati sono salvati in un database.

4.2 Requisiti di Sicurezza e Protezione

- I dati della rubrica devono essere protetti contro accessi non autorizzati utilizzando un sistema di password o crittografia.

4.3 Attributi di Qualità

4.3.1 QA Interni

- **Manutenibilità:** Il software della rubrica è scritto in un linguaggio chiaro (Java) e ben documentato, con commenti, in questo modo sarà più facile per uno sviluppatore esterno risolvere bug.
- **Modularità:** Il codice è scritto in maniera tale da avere una modularità alta, ci sono diverse classi con diversi metodi per far sì che ogni funzionalità sia a sé. Utilizzando questo approccio è più facile cambiare un metodo nella sua singolarità.
- **Efficienza del codice:** È stato utilizzato un algoritmo di ricerca basato sulle mappe in modo da rendere più efficiente la ricerca di un contatto all'interno della rubrica.
- **Riusabilità:** Il software è diviso in diverse classi, questo rende più facile la riusabilità di alcune parti del codice in contesti diversi.

4.3.2 QA Esterni

- **Usabilità:** Un utente può cercare un contatto facilmente e in maniera intuitiva cercando solamente il suo nome e/o cognome.
- **Prestazioni:** L'aggiunta, la modifica e la ricerca di un contatto sono tutte funzionalità della rubrica che impiegheranno meno di un secondo per compiere la loro azione.
- **Sicurezza (Safety):** I dati verranno salvati su un database in modo da non creare danni per la gestione delle informazioni dei contatti nella rubrica.
- **Sicurezza (Security):** I dati della rubrica devono essere protetti contro accessi non autorizzati utilizzando un sistema di password o crittografia.

Requisito	Priorità	Categoria	Breve descrizione	Rischio tecnico
L'interfaccia GUI	Alta	Usabilità	L'interfaccia utente deve essere visiva e basata su elementi grafici, facilitando l'interazione.	Basso
Formato numeri	Bassa	Affidabilità	I numeri di telefono devono rispettare un formato internazionale standard (es. +39 123 456 789).	Medio
Formato email	Bassa	Affidabilità	Gli indirizzi email devono essere validi secondo le specifiche RFC 5322.	Medio
Sicurezza dei Dati	Alta	Sicurezza	I dati della rubrica devono essere protetti contro accessi non autorizzati utilizzando un sistema di password o crittografia.	Basso
Limite Preferiti	Bassa	Usabilità	Il sistema deve permettere un massimo di 20 contatti nella lista dei preferiti.	Medio
Supporto Multilingua	Bassa	Accessibilità	L'interfaccia deve essere disponibile in almeno due lingue (es. italiano e inglese) e facilmente estensibile ad altre lingue.	Alto
Comportamento di Ricerca	Alta	Usabilità	La ricerca deve essere case-insensitive e supportare caratteri speciali.	Basso
Backup Automatico	Bassa	Affidabilità	Deve essere possibile configurare un backup automatico giornaliero o settimanale dei dati della rubrica.	Medio
Prestazioni Applicative	Alta	Prestazioni	L'applicazione deve rispondere alle operazioni comuni (es. ricerca, aggiunta, modifica) entro 1 secondo per il 95% dei casi.	Medio

Tabella 9: Categorizzazione dei Requisiti Non Funzionali